Fuzzy Logic in Autonomous Orbital Operations

Robert N. Lea
Software Technology Branch / PT4,
NASA / Johnson Space Center,
Houston, Texas

Yashvant Jani
Togai Infralogic, Inc.
Houston, Texas

ABSTRACT

Fuzzy logic can be used advantageously in autonomous orbital operations that require the capability of handling imprecise measurements from sensors. Several applications are under way at the Software Technology Laboratory, NASA / Johnson Space Center to investigate fuzzy logic approaches and develop guidance and control algorithms for autonomous orbital operations. Translational as well as rotational control of a spacecraft have been demonstrated using space shuttle simulations. An approach to a camera tracking system has been developed to support proximity operations and traffic management around space station Freedom. Pattern recognition and object identification algorithms currently under development will become part of this camera system at an appropriate level in the future. A concept to control environment and life support systems for large lunar-based crew quarters is also under development. Investigations in the area of reinforcement learning, utilizing neural networks combined with a fuzzy logic controller, are planned as a joint project with Ames Research Center.

KEYWORDS: ***autonomous orbital operations, fuzzy logic control, sensor fusion***

1. INTRODUCTION

The current activities of the Software Technology Branch of the Information Technology Division at the NASA Lyndon B. Johnson Space Center (JSC) are directed toward the development of fuzzy logic (Zadeh [1], Klir and Folger

Address correspondence to Robert N. Lea, Software Technology Branch / PT4, NASA / Johnson Space Center, Houston, TX.

655 Avenue of the Americas, New York, NY 10010 0888-613X/92/$5.00

[2]) software capabilities for building expert systems. In particular, the emphasis has been on developing intelligent control systems for space vehicles and robotics. The problem of sensor data monitoring and control of data processing, which includes detection of potential failures in the system and in some cases reconfiguration, is also under investigation. Results of performance tests made on simulated operational scenarios have been very promising. The issues of when, why, and how hardware implementation can be beneficial are also being studied carefully.

There are certain key technology utilization questions to be answered relative to the use of fuzzy logic control over conventional control.

1. **Is it possible to create control systems that do not require a high degree of redesign when system configurations change or operating environments differ?**

In other words, can adaptivity be achieved through the use of a fuzzy logic based controller in place of a conventional controller? Experience with the conventional controller development tells us that a typical conventional controller requires significant redesign when there are changes in (1) system characteristics, (2) system configuration, or (3) the environment in which the system is operating.

2. **Can a fuzzy controller be used as a high-level controller to function in conjunction with classical controllers in a way the human would?**

Specifically, can a high-level fuzzy controller be designed to monitor an existing system, evaluate its performance, and either suggest or force changes to make the system work properly or at least function more efficiently? A high-level controller typically works with abstract parameters that are derived but not directly measured. It also commands parameters other than direct control parameters. There are additional steps between the control function and the sensing as well as command functions. Such controllers are grouped as intelligent controllers (Antsaklis et al. [3]) and are not included in the conventional PID controllers group because these controllers perform additional tasks that provide capabilities for self-governing or regulation as well as fault tolerance.

3. **How easy or difficult is it to design and implement a fuzzy rule base that will control a complex system as opposed to developing a classical control system to solve the same problem?**

Fuzzy logic based controllers will be valuable in systems that are highly nonlinear and have complex environments that are practically impossible to model. Fuzzy controllers work for linear systems also but probably have less justification in this case, unless the problem is best thought of in a rule-based framework. The Japanese researchers and engineers have demonstrated (Rogers and Hoshai [4], Johnson [5], Armstrong and Gross [6]) the usefulness of fuzzy controllers in the last few years with some impressive applications from an engineering viewpoint, such as the Sendai train controller (Yasunobu and Miyamoto [7]), air-conditioning control systems, camera autofocusing systems (Shingu and Nishimori [8]), gas cooling plant controllers (Tobi et al. [9]), television autocontrast and brightness control, applications to automobile transmission and braking control, and applications to control of jitter in camera imaging, which requires distinguishing between real motion in the image, which is desired, and motion of the camera, which needs to be filtered out.

4. **Particular questions of interest to NASA are, Where can hardware implementations be utilized advantageously, and how easy or difficult is it to transfer fuzzy rule bases to hardware?**

In many cases, hardware will be able to take much of the computational burden off the central computing system. Fuzzy processors that perform fuzzy operations and execute fuzzy rule bases have emerged in the computer market (Yamakawa [10], Togai and Corder [11], Corder [12], Watanabe [13]) and are expected to gain widespread support for inline control of devices. Analog (Tasaka [14], Johnson [15]) as well as digital fuzzy processors are available and can be tailored to specific applications for optimum performance. Space operations can benefit greatly if the speed and power of these fuzzy processors can be utilized to achieve autonomy.

In Section 2, a typical mission scenario for autonomous orbital operations is described with activities and tasks involved in carrying out some important steps. The role of fuzzy logic in these operations is discussed in Section 3. A short summary of applications of fuzzy logic achieved thus far in the Software Technology Laboratory (STL) is provided in Section 4. Current activities that use fuzzy logic in orbital operations, future activities, and a summary of our approach are provided in Sections 5, 6, and 7, respectively.

2. AUTONOMOUS ORBITAL OPERATIONS

A typical rendezvous mission scenario as shown in Figure 1 for satellite servicing (Lea and Jani [16]) requires orbit transfers, rendezvous planning, phasing maneuvers, and guidance and targeting for proximity operations. These tasks are required to approach and capture a satellite for repair or maintenance or to return it to a space station or the Earth. Repair and maintenance of satellites also requires control of robotic manipulator arms if such repairs are to be performed at the satellite location as opposed to returning it to a permanently manned orbital facility or Earth. Sometimes a satellite may require only an inspection to determine if there are any problems. In this case, only stationkeeping or fly-around maneuvers are necessary.

In the problem of rendezvous of two space vehicles, it is typically assumed that the target vehicle can maintain a stable orbit during the time required for the rendezvous to take place. Ideally, it will also have a stable attitude

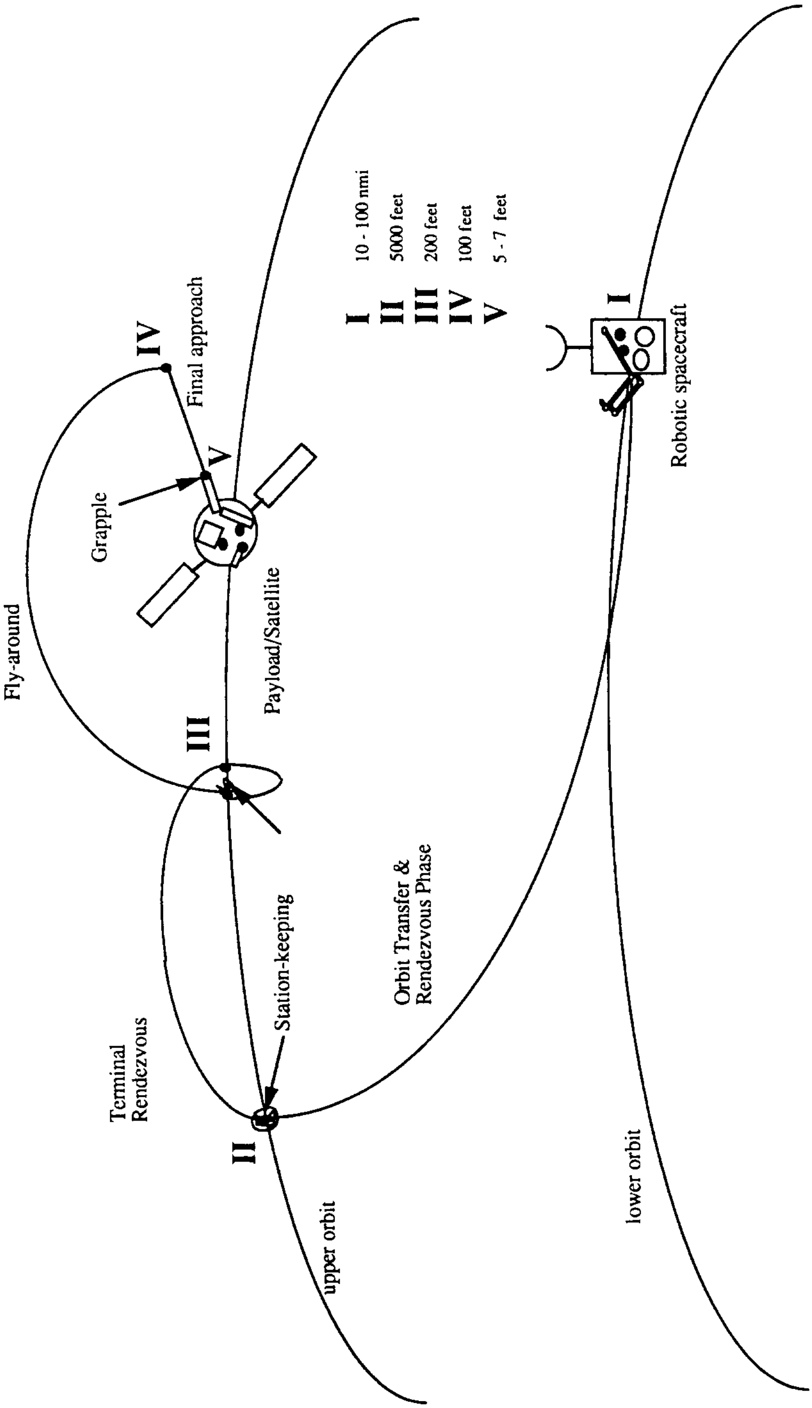


Figure 1. Typical satellite servicing mission scenario.

(although for vehicles in distress this may not be possible). For severely distressed vehicles, the actual orbit may also be affected. In either case, rendezvous and capture may be necessary.

The target vehicle will be assumed to be at the origin of a coordinate system known as the local vertical, local horizontal (LVLH), where positive z is directed from the target to the center of the Earth (or, in general, to the center of whatever body it is orbiting), positive y is along the negative of the angular momentum vector, and positive x completes the right-handed coordinate system, as shown in Figure 2. The chasing vehicle will be the only vehicle assumed to be able to intentionally modify its trajectory and attitude in this relative coordinate system. The performance of the tasks above require trajectory control of the active vehicle relative to the target vehicle, including not only relative positions of the two vehicles but also the attitude of the active vehicle.

Among the rendezvous mission tasks, mission planning based on mission goals and constraints is at the highest level. For example, a scenario for the capture of a satellite will incorporate time requirements, fuel constraints, and lighting and communications requirements based on the best assessment of the current and projected situations. The system will have to be intelligent enough to continually evaluate the status of the rendezvous and learn to adapt to unexpected occurrences through contingency planning or real-time tuning of control algorithms. Such a system will require many inputs from a variety of independent sources: ranging and visual sensors, navigation systems, object recognition systems, human inputs from ground-based or space-based stations, onboard planning systems, diagnostic systems that report on the health of various systems including individual sensors, and redundancy management systems. Some specific problems are tracking of moving objects with sensors such as cameras, radar, lasers, or star trackers. In the event of multiple objects in the vicinity of the desired target vehicle, it must be possible to recognize the proper one, and for final approach to the vehicle it will be necessary to recognize objects on the target vehicle such as docking ports or grapple fixtures.

The next important task is trajectory control, especially the control of relative position with respect to the target vehicle. This must be performed during the entire time of the rendezvous. In some segments, control has to be very precise, while in other segments the accuracy requirements are a bit relaxed. Trajectory control requires a continuous knowledge of current state, which is typically derived from several sensor measurements. It also requires the information regarding a desired state typically provided by the guidance systems. It should be noted that the information required for the trajectory control is continuously changing with time and is highly dependent on the accuracy of sensor measurements.

Similarly, attitude control is required throughout the mission. A robust attitude control system enhances trajectory control because the execution of

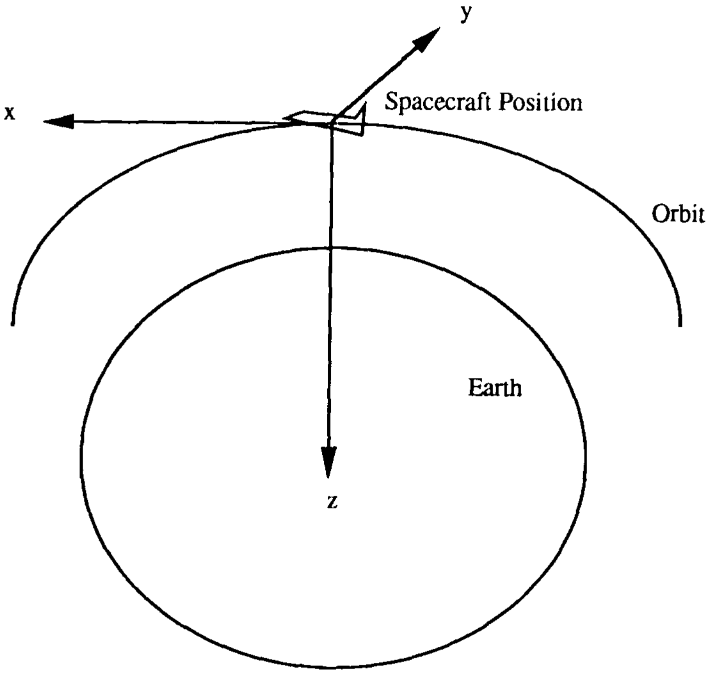


Figure 2. The local vertical, local horizontal coordinate frame.

desired delta-V is much more accurate. Poor attitude control can definitely result in a mission failure. It should be noted that rotational control has to be very precise during the final approach and docking segments because coupling between rotational changes and the relative distances is significantly high. Again, note that the knowledge regarding current as well as the desired attitude is required, and this information changes with time.

Both of the above tasks require processing and synthesis of sensor data. All measurements must be accurately interpreted, and action must be taken accordingly. Since several sensors are used, proper data fusion must be performed, and each measurement must be used in its proper context. Otherwise, the probability of mission failure increases very significantly. This data fusion task necessarily includes the monitoring task, which must be continuously performed, and any deviations from the planned trajectory must be reported immediately.

Once the chaser spacecraft gets close to the target, its approach to the docking port must be carefully maintained with tight control of both its translational and rotational states. The controller must have some provision for a recovery procedure in case of a docking failure. When the crew performs these functions, they interpret the measurements according to their training and take action according to the procedures developed in a simulator. These procedures typically include contingency steps in case a docking failure occurs. The autonomous vehicle must have the same capability for mission success.

The vehicle must prepare for return to base with or without the payload. These preparations could be very lengthy or very short depending on what procedure the crew decides to use and how their sequence of actions is organized. In any event, thinking like the crew will definitely help solve the problem of increasing autonomy in rendezvous operations.

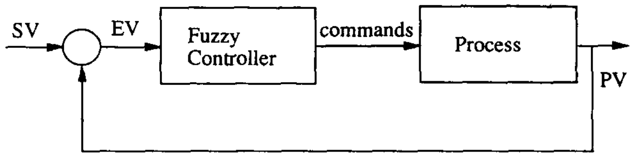
3. ROLE OF FUZZY LOGIC IN AUTONOMY

Fuzzy logic will be useful in the proper interpretation of measurements from sensors that are always corrupted by noise and bias. The accuracy of sensors represents a challenge that is not always surmountable. A fuzzy logic framework (Zadeh [1], Klir and Folger [2]) can easily handle imprecise measurements, thus helping the integration process. Also, systems may perform incorrectly or at least unexpectedly anomalously for a short time. It is necessary to determine this type of behavior and correctly resolve the situation. Processing of uncertain information using commonsense rules and natural language statements is possible in this fuzzy logic framework.

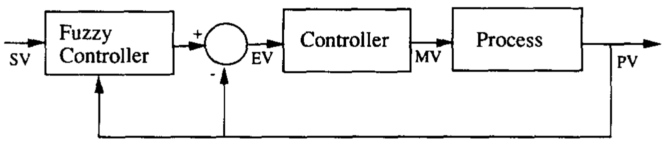
The utilization of sensor data in engineering control systems involves several tasks that historically are done by a human in the decision loop. These include cursory monitoring of data to determine if they should be processed and/or monitoring the output of the system to determine whether the system is performing as expected. All such tasks must be performed based on evaluations of the data according to a set of rules that the human expert has learned, usually from experience. Often, if not most of the time, these rules are not crisp, that is, there must be some commonsense or judgmental decisions made. Such problems can be addressed by a fuzzy set modeling approach, which if done properly can make decisions as well as the expert can.

The fuzzy logic approach is simple to understand and easy to implement as a software module. Fuzzy rules provide a framework to implement the human thinking process; that is, the rules reflect the human thought process, such as "If the object is Far_Left then rotate the camera to the left side." The entire rule base for the controller can be derived in the form of natural language statements as if a human were performing the controlling task. The experiential knowledge of a human controller, the crew in the case of space vehicles, can be easily embedded in the software.

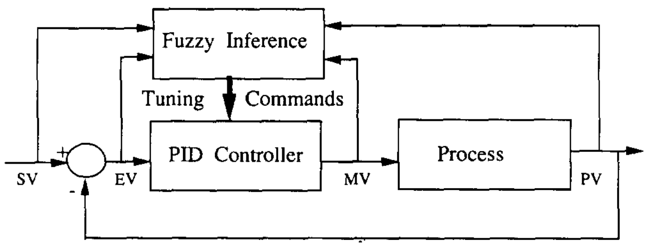
Fuzzy logic based controllers can be implemented in several ways as shown in Figure 3. In a strict sense these can be implemented as single controllers with well-defined input and output parameters. They can also be implemented



Fuzzy Controller



Fuzzy Feed-Forward Controller



Fuzzy Auto-Tuning System

Figure 3. Fuzzy logic based controllers in various configurations.

as feed-forward controllers in conjunction with conventional controllers such that the desired state-value can be altered to provide an appropriate correction. The final command for the process is generated by the conventional controller. An alternative is to implement the fuzzy controller as a tuning system (Togai [17]) in such a way that the parameters of a PID controller are tuned to better control the process and achieve efficiency. Thus, fuzzy logic controllers offer flexibility and adaptability for the process environment.

Implementation of fuzzy membership functions, rules, and related processing is made easy by tools like the TIL Shell (Perkins et al. [18], Hill et al. [19]), which has a graphics-oriented user interface and fuzzy-C compilers (Teichrow and Horstkotte [20]) that can generate code for a fuzzy chip or the C code to integrate with other software modules.

There are several commercial products available in the industry that allow easy implementation of knowledge base, rule base, and user interfaces. For autonomous operations, it is easier and more useful to implement control decisions through knowledge bases and rules so that the heuristics and related experiential knowledge can be used for a particular situation.

It is also possible to develop and implement a fuzzy controller in a fuzzy processor, thus having a fuzzy hardware controller. There are several commercial fuzzy processors (Yamakawa [10], Togai and Corder [11], Corder [12], Watanabe [13], Tasaka [14], Johnson [15]) that can process over 30,000 fuzzy rules per second and thus provide a high processing power. These fuzzy processors consume low power with a capability to process general-purpose instructions and can be mounted in the back plane of a sensor—for example, a camera. These processors also provide interfaces to hardware as well as the main computer to transfer information and commands. The advanced sensor systems envisioned for space station operations will have such processors embedded as an integral part of the system. Thus, a distributed processing system on board the spacecraft is possible via fuzzy chips.

A camera tracking system (Lea et al. [21]), described in Section 4.4, can be a dedicated sensor with built-in intelligence and speed to perform functions that will normally be performed by the onboard computers. Because of the dedicated nature of a fuzzy chip and its processing power, there will be virtually no computational load to the space station *Freedom* computers. As a result, the computers will be available for other computing requirements such as complex guidance and navigation schemes. Furthermore, the interfaces between the fuzzy chip and computers will be at a command level requiring reasonably low speed data transfer. There will be no need for a high rate of data transfer, that could possibly increase costs and decrease reliability.

A significant application area of fuzzy logic is in an advisory role in health monitoring and internal reconfiguration of spacecraft subsystems. These processes require a capability to handle uncertain measurements, estimate possibilities of failures, and quickly rearrange flow so that autonomous operations are not stopped. Techniques have been developed to update the rule base using reinforcement learning in a given environment and to adjust the response or behavior of a controller. These techniques are very important for achieving operational efficiency in space operations.

4. PAST ACCOMPLISHMENTS AT JSC

There have been several applications of fuzzy logic to orbital operations at JSC. Sensor data processing control for star tracker navigation evolved during 1985–1986 (Lea and coworkers [22–26]) and was successfully demonstrated to analyze space shuttle rendezvous flight data. A translational control spacecraft system based on fuzzy rules (Lea [27, 28]) was developed during 1987–1988 and demonstrated (Lea [29]) during the International Fuzzy Systems Association (IFSA) video teleconference IFSA88 at Iizuka, Japan, in 1988. Rotational control for spacecraft attitude maintenance has been developed (Lea and Jani [30]) using the phase-plane approach and was demonstrated at the IFSA89 conference (Lea et al. [31]) in Seattle in 1989. A fuzzy logic based concept for a camera tracking system has been developed and was reported at the Eighth International Congress of Cybernetics and Systems in June 1990 (Lea et al. [21]). These applications are described in a short summary in this section.

4.1. Sensor Data Processing

In space shuttle rendezvous operations, the star tracker is used to give angle measurements for tracking rendezvous targets when the sun–target–shuttle geometry is such that the target is reflecting sunlight toward the shuttle star tracker and when radar data are not available. Certain problems can occur when attempting to track a target with the star tracker. For example, a star or debris such as ice crystals (caused by shuttle venting or jet firings) may be acquired. Loss of lock on the true target and subsequent reacquisition of a false target are possible, especially if the target is dim due to attitude or range or if the target is tumbling. If a bright false object crosses the target line of sight (LOS), the star tracker is likely to follow the brighter object. From experience, using simulated data, we know that the shuttle rendezvous navigation filter will process data under these conditions for a long enough time that the state vector will be destroyed.

Under current operational conditions, to guard against any of these problems, a crew member monitors the acquired signal for acceptability prior to allowing the data to be processed, and he or she monitors the residuals during data processing to ensure that no unusual problems occur. To determine acceptability for processing, the shuttle crewman observes the following rule (Lea [25]):

> If the residual is *less than* the expected error as determined in pre-mission studies, and the *change* in residuals is *less than* 0.05° for *five* consecutive measurements, then allow the filter to process data.

This rule contains deterministic conditions that are actually *fuzzy* in nature and have been interpreted as fuzzy by the crew at times during actual operations. The general problem considered here is to model the crew member's reasoning and commonsense thought process in deciding whether the sensor data are acceptable for use in updating the shuttle–target relative state vector. This involves preediting and screening the data and weighting the relative state vector update.

The Kalman filter editor, as designed for the shuttle rendezvous navigation system, compares the residual magnitude against a multiple of the expected variance in the residual as derived from the current covariance matrix and the expected sensor error model. Data for which the residual is less than or equal to the expected error are incorporated into the filter state, and data for which the residual exceeds the expected error are not processed by the filter but are displayed to the crew for use in decision processes.

The filter and editor have performed satisfactorily on all rendezvous flights thus far. However, it has been considered essential that the crew be involved in the operations, or else erroneous data, such as those obtained from locking onto stars and debris, will be processed by the filter, thus corrupting the filter state and necessitating a filter restart. With the current editor design it is not possible to protect against this because a star or debris may be very close to the target LOS.

The crew preediting function is to ensure that the true target is acquired prior to data processing. If the object acquired is the true target, the residual should be less than the expected error, but more important it should stay almost constant. The only variation should be from noise in the sensor and small errors due to propagation of the shuttle and target states. Residual change less than 0.05° is a conservative requirement consistent with the noise and bias in the star tracker.

Star tracker data are useful in maintaining a good relative state vector, but since it gives no range information directly, the state vector is easily corrupted by erroneous data. To guard against processing erroneous data, and to ensure that good data are processed, two things have been done. First, the preediting rule has been restated using fuzzy sets, which seems more appropriate than crisp statements in terms of ordinary Boolean logic. The fuzzy variation of the rule (Lea [25]) reads as follows:

> If the measurement *residual* is *small* with respect to the expected value as determined from pre-mission studies and the *residual change* is *small* with respect to expected propagation errors and noise in the sensor for *several consecutive* measurements, then allow the Kalman filter to process data.

Second, so that the measurements will be processed in a way consistent with commonsense reasoning, the decision function for processing data was modeled as a fuzzy set to be used for weighting updates to the state vector. By doing this it is ensured that measurements that are close together will be processed similarly. For example, a measurement that slightly passes the residual edit criterion and one that barely fails will be processed similarly—the one that slightly passes will be allowed to contribute only slightly to the state vector update.

In summary, if $\vec{x}$ is the state vector, **W** the Kalman filter weighting matrix, $\vec{r}$ the measurement residual, and f the fuzzy decision function, then the update to the state vector is given by

$$\vec{x} = \vec{x} + f(\vec{r}) * W\vec{r}$$

Generally, as the filter converges, $f(\vec{r})$ approaches 1 and the update process converges to the ordinary Kalman filter equations.

In order to implement the decision rules, fuzzy sets are used. The notion of intersection as defined by Zadeh [1] has been used. The definition is given here for completeness. If f and g are fuzzy sets, then

$$(f \wedge g)(x) = \min\{f(x), g(x)\}$$

Now, if we let p, q, and r be fuzzy sets representing *residual is small*, *residual change is small*, and *residual change is small for several measurements*, respectively, then the preediting and weighting rule can be restated as follows.

> If $p \wedge r$ is greater than some tolerance, then update the state vector after downweighting with factor $p \wedge q$.

In the rule the magnitude of $p \wedge r$ controls the decision for editing or inhibiting data. One can place restrictive conditions on the decision function; if $p \wedge r <$ 0.5, then edit the measurement completely. In this study different values were tested in an effort to determine if the rule is sensitive to different edit levels, and it was found that because of downweighting factor it worked as well as to set the edit level to 0.0.

There were also other rules that had to be considered for this data control system. They are not discussed here in detail, but it is obviously necessary to force break lock and reacquisition when there are extended periods of edited or severely downweighted updates. It clearly does no good to have the sensor supplying bad data. These rules were included. Simply stated, they say that if the edit decision function is less than a preset tolerance for a preset number of consecutive measurements, then break lock and reacquire.

Details of the type function used for p, q, and r are given by Lea and Goodwin [22] and Lea [23]. Lea and Giarratano [24] reported studies that indicated that the type and shapes of fuzzy functions used for these data-editing problems are not critical to success.

The fuzzy editing criterion was implemented into a simulation version of the shuttle onboard software. Real mission data were processed through this simulation, and inputs to the filter were controlled by the fuzzy decision-making process defined by the rule rather than by the crew and the current filter editor. The data from this simulation were compared to the results obtained during actual missions under ideal conditions as determined by the crew.

For nonnominal flight data collected from the Solar Maximum Mission (SMM), the first shuttle rendezvous mission, the performance of the fuzzy editing scheme did not differ significantly from the current onboard system with the crew performing their normal preediting and monitoring functions. On this flight, an inertial measurement unit (IMU) problem caused apparent errors in the star tracker measurements that exceeded the expected error by a factor of about 50 (Lea [26]). The problem, although actually caused by the IMU redundancy management function, had the net effect of an extremely large random sensor measurement bias. These flight data also had measurements obtained from locking onto stars at the beginning of each star tracker interval. Instead of simulating a break track that would normally be done when data from false targets are acquired, it was decided to process the star data to test the weighting function's ability to handle problem measurements. As the data tabulated in Table 1 indicate, the erroneous data caused no problems (Lea [25]).

The state vector obtained using the fuzzy logic process and the state vector obtained from the actual flight data were then propagated for approximately 1 h until radar data were obtained. The two propagated vectors were compared to the radar data to evaluate the filter's performance with the fuzzy editing and weighting rule against the system performance with a human in the loop. For this test the $p^{\wedge}r$ edit level was set to 0.0. The range and range rate estimates from the onboard navigation system and the system with fuzzy editing and weighting, after propagating the state vectors for 1 h, are then compared to the range and range rate measurements from the radar. The deviations from the radar measurements for the two systems are approximately the same. For a radar range of 102,695 ft, the range deviations for the fuzzy data processing system and the onboard filter are 1965 and 1835 ft, respectively (Lea [25]). This fuzzy editing and process control application has thus given very satisfactory results, comparable to that achieved by the crew in the operational system.

4.2. Translational Control of a Spacecraft

Fuzzy sets have been used in developing a trajectory controller for spacecraft applications in proximity operations profiles (Lea [27, 28]). An automated vehicle controller that interprets the sensor measurements in a manner similar to that of a human expert has been modeled using fuzzy sets. The control rules were derived from the thinking process used by pilots and were implemented using typical π and s functions (Fig. 4) that can be adjusted for various degrees of fuzziness. Membership function and universe of discourse definitions were based on the targeting equations and control strategy for the LOS approach (Lineberry et al. [32]). The control strategy relied heavily on the experience base for manual operations.

Typical rules used for rendezvous vehicle control and modeled with fuzzy sets are the following:

> If the rendezvous vehicle's orientation with respect to a desired pointing vector to the target vehicle is close to the required orientation, then no action is necessary.If the orientation significantly deviates from the required, then take appropriate action to correct the problem.

Both in-plane and out-of-plane positions and rates must be controlled, as well as range and range rate. Fuzzy sets are defined for "somewhat greater than,"

Table 1. SMM Rendezvous Errors in Range and Range Rate Caused by IMU Switching (as Compared to the Onboard Solution) After Processing 20 Min of Star Tracker Data

	Range	Range Rate
Nominal filter (no star protection)	– 13,600	– 0.5
Fuzzy editor (editing when $p \wedge r = 0$)	300	0.31
Fuzzy editor (editing when $p \wedge r < 0.25$)	1,320	– 0.15
Fuzzy editor (editing when $p \wedge r < 0.5$)	1,150	– 0.11

"somewhat less than," and "approximately equal to" the desired closing rate. They are also defined for "high," "low," and "near" with respect to the desired position (see Fig. 4). During some time interval (every 2 s for the shuttle) the fuzzy sets are evaluated, and a determination is made as to whether an action should be taken to restore a rate or position to its desired value. If the no-change function, such as "approximately equal to" or "near" the desired value, is larger than the corresponding change function, such as "somewhat greater than" or "low" with respect to the desired, then no action is taken. Otherwise an appropriate action is taken to restore the rate or position to the desired. The appropriate action is determined from an estimated action $A(\mu)$, where μ is the current value of the state, required to restore the active vehicle to the desired position from some maximum expected deviation. This action

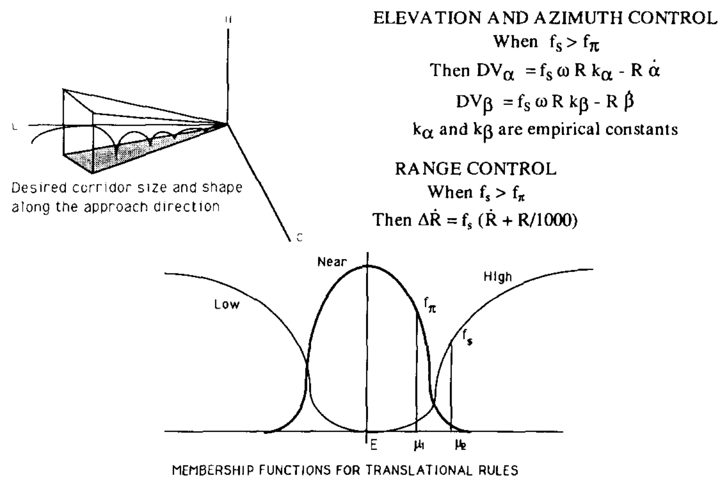


Figure 4. The π and s functions used in the control strategy for range parameter. For a given value of parameter μ, f_π is compared with f_s, and the larger value dictates whether action is taken.

$A(\mu)$ is then weighted by the change function $S(\mu)$, and the system under control is commanded to take an action $S(\mu)*A(\mu)$. Furthermore, there are no extreme accuracy requirements for the function $A(\mu)$. For example, referring to Figure 4, if μ_1 is the current value of x, then $\pi(\mu_1) > S(\mu_1)$, and no action is taken. On the other hand, if μ_2 is the current value of x, then $S(\mu_2) > \pi(\mu_2)$, and the command is given to take action $S(\mu_2)*A(\mu_2)$. More than one action can be ordered at a time as long as a constraint of the system under control is not violated.

The fuzzy controller has been implemented into a multivehicle dynamical simulator known as the Orbital Operations Simulator (Edwards and Bailey [33]), complete with all environment and sensor models. A small part of this control simulation was demonstrated via televideo links (Lea [29]) to the IFSA88 Workshop that was held in Iizuka, Fukuoka, Japan, in August 1988. In this simulation, the automated fuzzy controller was used to control the closing rates and relative positions of the shuttle with respect to the SMM satellite. According to the test scenario, the fuzzy controller was required to perform operations including approach to target, fly-around, and stationkeeping.

Many different scenarios have been run with this automated fuzzy controller to evaluate the performance with respect to flight profiles and delta-V requirements, which is a direct measure of the performance. Comparisons of delta-V requirements for a human-in-the-loop versus the automated controller showed (Lea [27]) that the automated controller always used less delta-V. For a test case involving stationkeeping at 150 ft for 30 min, the automated controller required 0.1 ft/s delta-V, whereas 0.54 ft/s was used in the human-in-the-loop simulation. For approach along the positive LVLH x axis, referred to as v-bar approach, from 500 ft to 40 ft within a 25-min time interval, the automated controller used 2.12 ft/s vs. 2.99 ft/s for the human-in-the-loop simulation.

4.3. Rotational Control of Spacecraft

To complement this translational controller, it was decided to implement a rotational controller via fuzzy membership functions and rules based on the conventional phase plane. It was obvious that such an implementation would provide a direct performance comparison with the conventional control system, thus leading to further insight into understanding the relative merits of fuzzy control systems. Furthermore, an integrated six degrees-of-freedom (6-DOF) controller can be developed by combining these two control systems.

The rotational control system has been developed on a 386 computer using the fuzzy-C compiler and related software (Perkins et al. [18]). The Phase_Plane package that contains the membership functions and rules has been implemented in a file called Phase.til (Lea and Jani [30]). The angle and rate errors, PHI and Phi_Dot, are inputs, and desired firing pulse level is the output for this rotational controller. The input variables have seven membership functions defined over the universe of discourse as shown in Figure 5. The output variable has five membership functions as shown in Figure 6. There are 25 rules defined for reducing the Phi and Phi_Dot errors to within their zero (ZO) range (see Table 2). These rules are based on the phase-plane construct used in the attitude control system.

Single-axis rotational equations were implemented for the pitch axis of the shuttle. The pitch moment of inertia and the positive and negative pitch torques based on pulse level value were used in this simple simulation to test the fuzzy controller rules. The shuttle jets provide a larger acceleration for positive pitch than for negative pitch. The simulation was set up to provide a constant torque during a cycle time of 80 ms. The pitch attitude and the rate are propagated at this cycle time. When the pulse level from the fuzzy controller is greater than 1, constant torque is provided in that direction; otherwise no torque is provided. This simulates the jet-on and jet-off activity at the appropriate time. The fuzzy controller is called at every cycle to evaluate all rules and output the desired firing pulse level. Based on this desired firing pulse level, the jet is turned on and the rate and angle are propagated. With new values of angle and rate, the angle error and rate error are computed for the next cycle input. Time

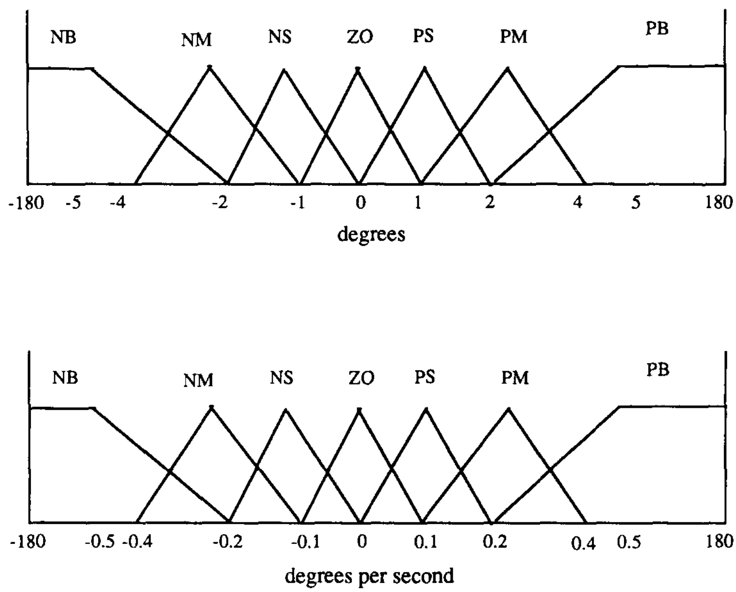


Figure 5. Membership functions for Phi and Phi_Dot. NB, negative big; NM, negative medium; NS, negative small; ZO, zero; PS, positive small; PM, positive medium; PB, positive big.

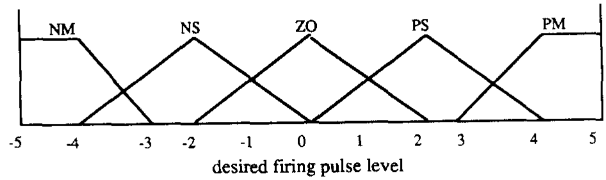


Figure 6. Membership functions for rotational acceleration. Abbreviations as in Figure 5.

is also advanced every cycle. Time histories of angle and rate and the phase-plane plot are created for analysis.

Testing for the pitch axis has so far shown very satisfactory results. With several starting states, that is, initial angle and rate, the system has converged on the commanded value and manifested a relatively smooth limit cycle around the deadband. The control system response in all cases has been as expected, including overshoot behavior in cases where initial rate error is very large. Tests were performed with some rules turned off or deactivated to observe the performance with a limited rule base. The objective was to reduce the number of rules to a minimum.

Performance of the fuzzy controller with 25 rules was more than adequate for a single axis and gave us confidence to expand it to the three-axis case. The phase-plane module in the shuttle digital autopilot (DAP) was replaced by this controller with all other interfaces unchanged. The integration process was completed with only minor modifications to the interfaces. The simulation testing included three-axis attitude hold and single-axis maneuvers. In a three-axis attitude hold case, the fuzzy logic based controller used only 30% as much fuel as the DAP. For the case of attitude maneuvers, the fuzzy controller

Table 2. Rule Base for Attitude Controller

	Phi						
Phi_Dot	NB	NM	NS	ZO	PS	PM	PB
NB	PM	PM	PS				
NM	PM	PM	PS				
NS	PS	PS	PS				
ZO	PS	PS	ZO	ZO	ZO	NS	NS
PS					NS	NS	NS
PM					NS	NM	NM
PB					NS	NM	NM

NB, negative big; NM, negative medium; NS, negative small; ZO, zero; PS, positive small; PM, positive medium; PB, positive big.

used around 60% as much fuel as the DAP. In both cases, the fuzzy controller showed a comparable performance for maintaining attitude and body rates. Detailed testing and analysis is planned to include other maneuver modes and different parameter sets.

4.4. Camera Tracking Control System

Advanced sensor systems with intelligence and a distributed nature will be required for activities like proximity operations and traffic control around the space station *Freedom*. There will be several sensors of different types providing various measurements simultaneously as input for processing to such a system. Conceptual development of such a system (Lea et al. [21]) where several cameras, laser range finders, and radar can be used as independent components is in progress within the STL at JSC. The first phase of this development is the camera tracking system based on the fuzzy logic approach that utilizes the object's pixel position as input and controls the gimbal drives to keep the object in the field of view (FOV) of the camera as shown in Figure 7.

Tracking an object means aligning the pointing axis of a camera along the object's line of sight (LOS). The monitoring camera is typically mounted on the pan and tilt gimbal drives, which are capable of rotating the pointing axis within a certain range. The task of the tracking controller is to command these gimbal drives so that the pointing axis of the camera is along the LOS vector that is estimated from the measurements.

For the fuzzy logic based tracking controller, the inputs are range and LOS vector, and the outputs are the commanded pan and tilt rates. The LOS vector is input in terms of pixel position in the camera FOV. When an image is received, it is processed to determine the location of the object in the camera frame, which has the vertical, horizontal, and pointing vectors as three axes. Usually, particularly for complex objects, an image spans many pixels. Using a suitable technique, the centroid of the image is computed and used as the current location of the object in the viewing plane. This plane is a Cartesian coordinate plane having vertical and horizontal axes. The size of the viewing plane is 170×170 pixels, and the origin is at the upper left corner as shown in Figure 7. The range of the object is received from the laser range finder as a measurement. These three parameter values are input to the controller.

Membership functions for the range, horizontal, and vertical positions are shown in Figure 8, and the membership functions for the Scale_Factor, Pan, and Tilt rates are shown in Figure 9. For simplicity, these functions are triangular over the universe of discourse. The Scale_Factor parameter is used as an intermediate step and provides the desired flexibility of changing the responsiveness of the fuzzy controller.

The desired image location is the center of the viewing plane, which is at

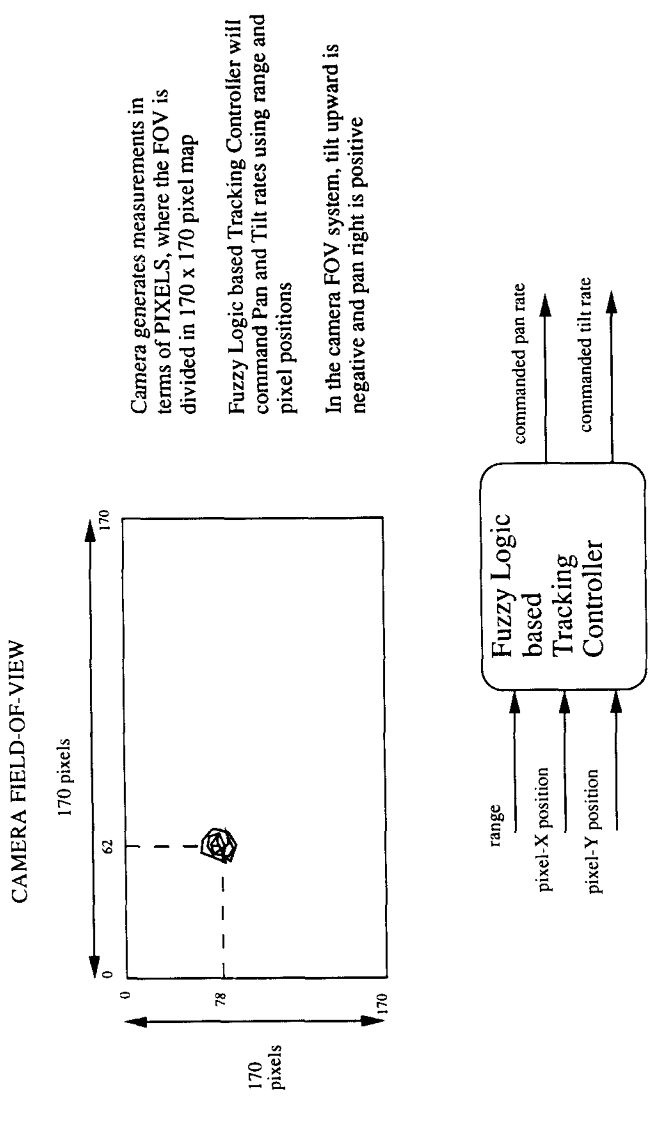


Figure 7. Concept of a camera tracking system.

(85, 85). If the current location is close to the center, then rotation of the pointing axis is not required. If the location is to the left of center, then a left rotation is necessary. Similarly, if the image is down from the horizontal line, then a downward rotation is required. These rotations are determined using the position and range measurements and the rule base shown in Table 3. First the range measurement is fuzzified and the value of the scale factor is determined

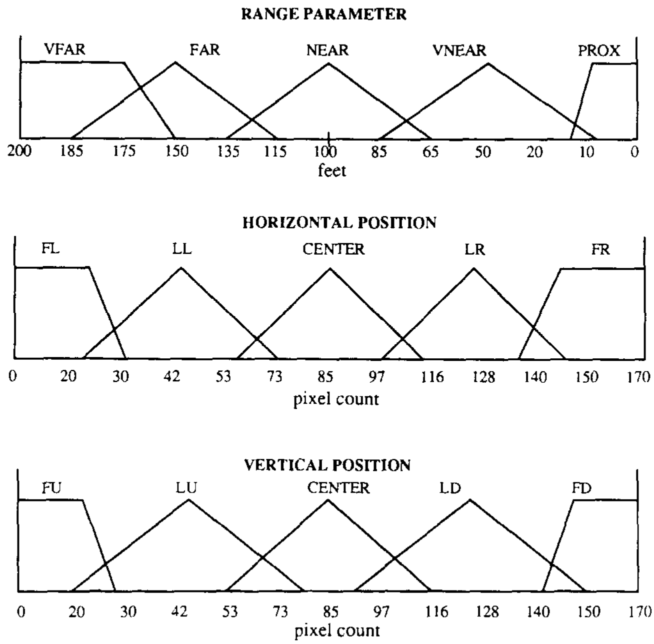


Figure 8. Membership functions for input parameters. VFAR, very far; VNEAR, very near; PROX, proximity zone; FL, far left; LL, little left; LR, little right; FR, far right; FU, far up; LU, little up; LD, little down; FD, far down.

based on the Scale_Factor rules. Necessary defuzzification processing is performed to compute the crisp value of the scale factor. Then the defuzzified scale factor, from the execution of the previous rules, and the position measurements are provided to the next set of rules to determine the rate at which the gimbal drives should be rotated. The defuzzified output of these 30 rules are the desired pan and tilt rates and are sent to the gimbal drives as command values.

The camera is rotated based on these commands within the limits of its gimbal rates and angles. New LOS measurements in the camera FOV are obtained for the next cycle, and the processing is repeated. The cycle time is based on the processing time required for the following functions: (1) determining pixel positions, (2) obtaining a range measurement, (3) rotating the gimbal drives at a desired rate, and (4) the requirements to track the object within a certain performance envelope. Typical cycle time ranges between 0.1 and 1.0 s.

There are several advantages to be gained in the development of a camera tracking system. This system will involve low-power sensors as compared to active sensors, for example, radar in the Ku band range, or radar using laser

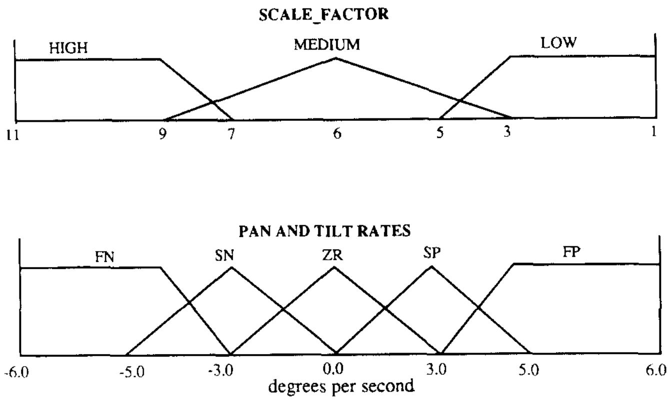


Figure 9. Membership functions for Scale_Factor and Output parameters for camera tracking system. FN, fast negative; SN, slow negative; ZR, zero; FP, fast positive; SP, slow positive.

frequency. Typically, the active sensor radiates a power pulse toward a target and receives back a reflected pulse. Based on the power transmitted, power received, and time between the pulses, parameters such as range and range rates are calculated. Since the camera tracking system will not be radiating

Table 3. Rule Base for the Tracking Task[a]

	Distance Membership Functions				
	VFAR	FAR	NEAR	VNEAR	PROX
Scale_Factor	LOW	LOW	MED	HIGH	HIGH
	Horizontal Position Membership Functions				
Scale_Factor	FL	LL	CENTER	LR	FR
LOW	FN	SN	ZR	SP	FP
MED	SN	SN	ZR	SP	SP
HIGH	SN	ZR	ZR	ZR	SP
	Pan_Rate Membership Functions				
	Vertical Position Membership Functions				
Scale_Factor	FD	LD	CENTER	LU	FU
LOW	FP	SP	ZR	SN	FN
MED	SP	SP	ZR	SN	SN
HIGH	SP	ZR	ZR	ZR	SN
	Tilt_Rate Membership Functions				

[a]Negative Tilt_Rate means the pointing axis going upward in FOV.

power, it will require low power in comparison with active sensor systems. Because there is already a shortage of power, an important consumable, on board the space station *Freedom*, availability of low-power sensors is very important for continuous operations. The SSF can afford to keep this type of sensor working around the clock without having much impact on the power management or other computational load on the main computers.

5. CURRENT ACTIVITIES

In this section, we describe the current ongoing activities in STL in the area of fuzzy logic research. A complete 6-DOF controller is being created by combining the translational and rotational controllers. Our integration approach and testing philosophy are described in Section 5.1. Our plans for software and hardware testing for the camera tracking system are described in Section 5.2. Activity in the area of motion control for a Mars rover vehicle during the sample collection process is described in Section 5.3 along with some preliminary results.

5.1. Combined Translational and Rotational Control for Relative Orientation and Distances

The integration approach adopted for combining translational and rotational control systems is simple and straightforward and involves extensive testing (Lea et al. [31]). The first step is to implement the previously defined translational rules in the same format using our development environment. This will provide commonality between the codes and allow an opportunity for stand-alone testing and optimization of translational rules. The second step is to generate the proper code for the SUN workstation using the fuzzy-C compiler (with appropriate options) and transfer it to the workstation. This step is required only because the development environment is on the 386 computer and the high-fidelity simulation is on a SUN workstation. Since the fuzzy-C compiler and associated development environment is portable, there is a plan to develop fuzzy controllers on the SUN workstation and avoid the code transfer. The third step is to develop the test plan that will test all aspects of the 6-DOF controller. The final step is to perform testing and compare the results with those of the conventional system.

NASA's Orbital Operations Simulator (OOS) (Edwards and Bailey [33]) will be used for testing the 6-DOF controller. It is a high-fidelity multivehicle spacecraft operations simulation that provides 6-DOF equations of motion within an orbital environment that includes aerodynamic drag. It can be used for engineering analysis as well as real-time operations demonstrations. It provides a framework to integrate and test expert systems and hardware with the software modules commonly known as onboard flight software. The OOS

(Fig. 10) executive also provides external interfaces to graphics and expert systems.

The translational fuzzy control system (Lea [27]) will be used by the autosequencer to generate proper hand controller commands so that the desired range and range rate are maintained during proximity operations. Typically, a shuttle pilot provides these inputs and controls the relative trajectory. Thus the autosequencer will simulate the crew input via the translational fuzzy control system. The automatic attitude control system of the shuttle on-orbit digital autopilot (DAP) is implemented in the OOS for shuttle on-orbit operations. The rotational fuzzy control system created by replacing the phase-plane module will generate commands for jet-select to fire jets for attitude control. Existing interfaces with the phase-plane module will be maintained intact for the overall integrity of the system. When both fuzzy control systems are used together, it will provide a total 6-DOF controller for proximity operations.

A preliminary test plan has been put together to test the 6-DOF controller. It includes test cases for stationkeeping with a fixed attitude, stationkeeping with attitude changes, LOS approach on the V-bar, LOS approach on the R-bar, fly-around at a constant distance with constant relative attitude, and final approach for docking. Details of these test cases such as initial conditions and commanded attitude maneuvers are being defined to finalize the test plan.

5.2. Implementation of Fuzzy Controller for a Camera Tracking System

Activities planned for this year for the camera tracking system include testing of the concept in software as well as hardware simulations. The software testing will be performed in the STL using a 386-based system as well as SUN workstations. The hardware testing will be performed in collaboration with the Engineering Directorate at JSC. It should be emphasized that the software testing will help fine tune the rule base and the membership functions, while the hardware testing will help to identify all interface problems, evaluate real-time performance, and fine tune the controller in light of actual measurements, which will be noisy. Both software and hardware testing are required to make the system operational and useful.

The tracking controller described in Section 4.4 has been implemented using the fuzzy-C development system, and necessary software modules in C language have been generated. Its interfaces with the sensor module that provides the measurements and the gimbal drive module that accepts the commands have been defined and implemented in C. A top-level executive has been designed as shown in Figure 11 with the necessary data flow and the state propagator for a target vehicle. At this time, the Clohessy–Wiltshire equations of motion (Rockwell International [34]) in the LVLH frame will be used to propagate the target state and to generate the camera measurements. A first-order linear gimbal drive model has been developed for the pan and tilt

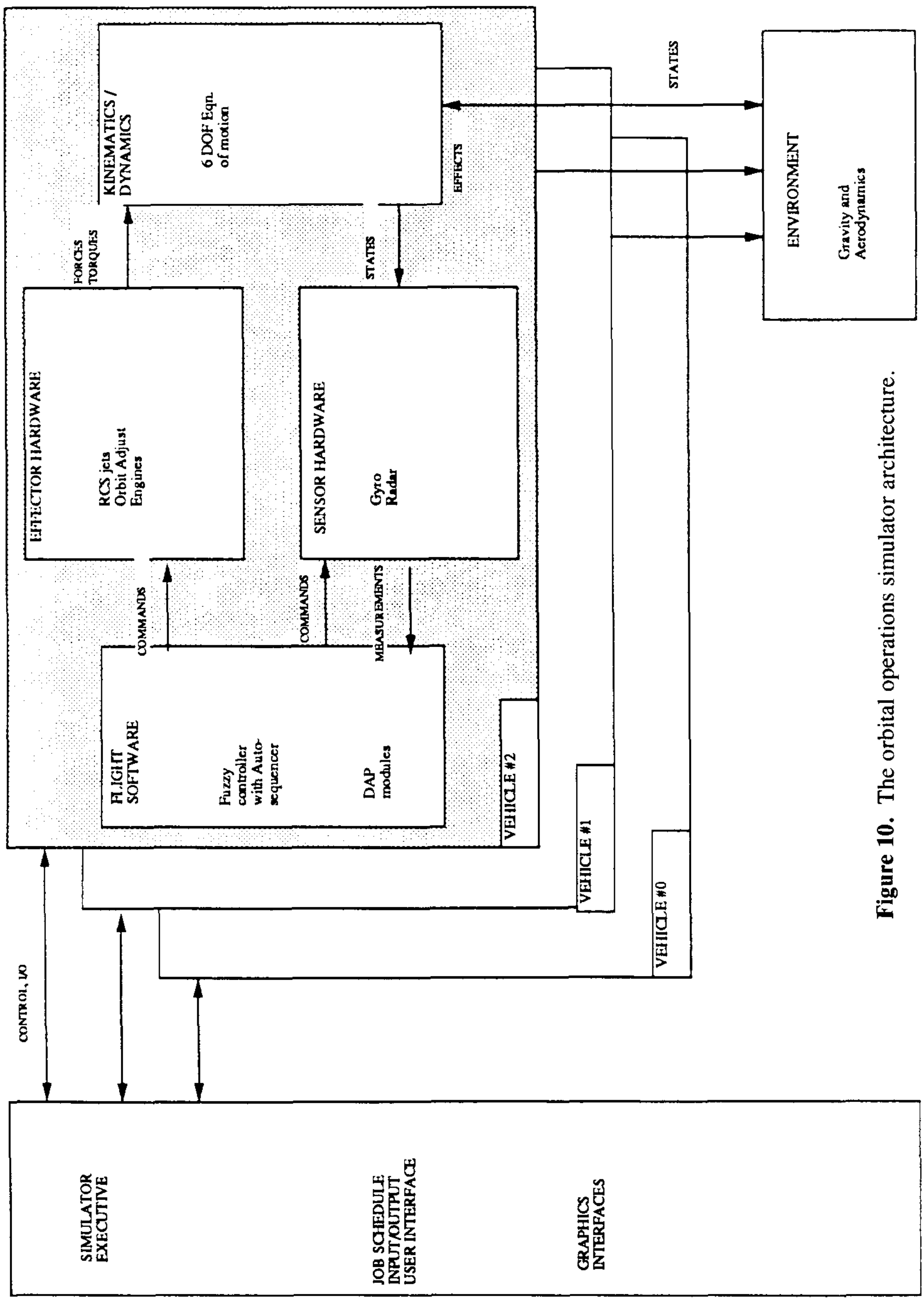


Figure 10. The orbital operations simulator architecture.

servo drives to rotate the pointing axis of the camera. The measurements for the range and horizontal and vertical positions are based on the geometry in an LVLH frame. A detailed test plan will be defined to test the concept for several different scenarios. The fuzzy tracking controller will be tested for the following types of relative trajectories: approach, fly-around, stationkeeping, and passing orbits.

The hardware laboratory in the Engineering Directorate has the necessary equipment for testing: camera, gimbal drives, laser range finder, and other interface equipment. The camera system will require a digitizer or pixel map generator and interfaces to the computer. The fuzzy controller software developed at STL will be ported to this computer, which will have the necessary hardware interfaces. A test plan that includes real moving targets in the laboratory and various lighting conditions to simulate the orbital environment will be generated, and the performance of the fuzzy controller will be analyzed in detail. A study will be performed to determine the responsiveness of the gimbal drives with respect to the changing Scale_Factor membership functions.

There is considerable effort at STL devoted to the development of algorithms for object identification and pattern recognition. In particular, emphasis is given to algorithms for performing scene analysis and extracting the information from the image using fuzziness and related parameters (Pal [35]). Results of this effort can be implemented and integrated at various levels in the concept of the camera tracking system to extend its capabilities to include image processing. How to integrate these algorithms and at what level will be investigated as part of our current activities.

5.3. Trajectory Control for Mars Rover During Sample Collection

While collecting soil samples and surveying the Mars surface, the Mars rover will be moving from one point to another among obstacles that cannot be identified prior to the mission. To complete the collection task, the rover must interpret imprecise sensor measurements of obstacle size and distance to determine which obstacles present a hazard and must be avoided and to replan trajectories to avoid these unforeseen obstacles as they are observed. In addition, since the worst-case round-trip communications time between Earth and Mars is 20 min, Earth-based telerobotic control of the Mars rover will be extremely difficult and time-consuming and could seriously endanger the success of the mission. Fuzzy trajectory planning and control provides robust real-time control capable of adapting the trajectory profile to avoid unforeseen hazards. The fuzzy logic approach eliminates communications travel time, allows the rover to avoid obstacles that may be unavoidable with telerobotic operations due to reaction time, and provides adaptable control that will extend the rover performance envelope.

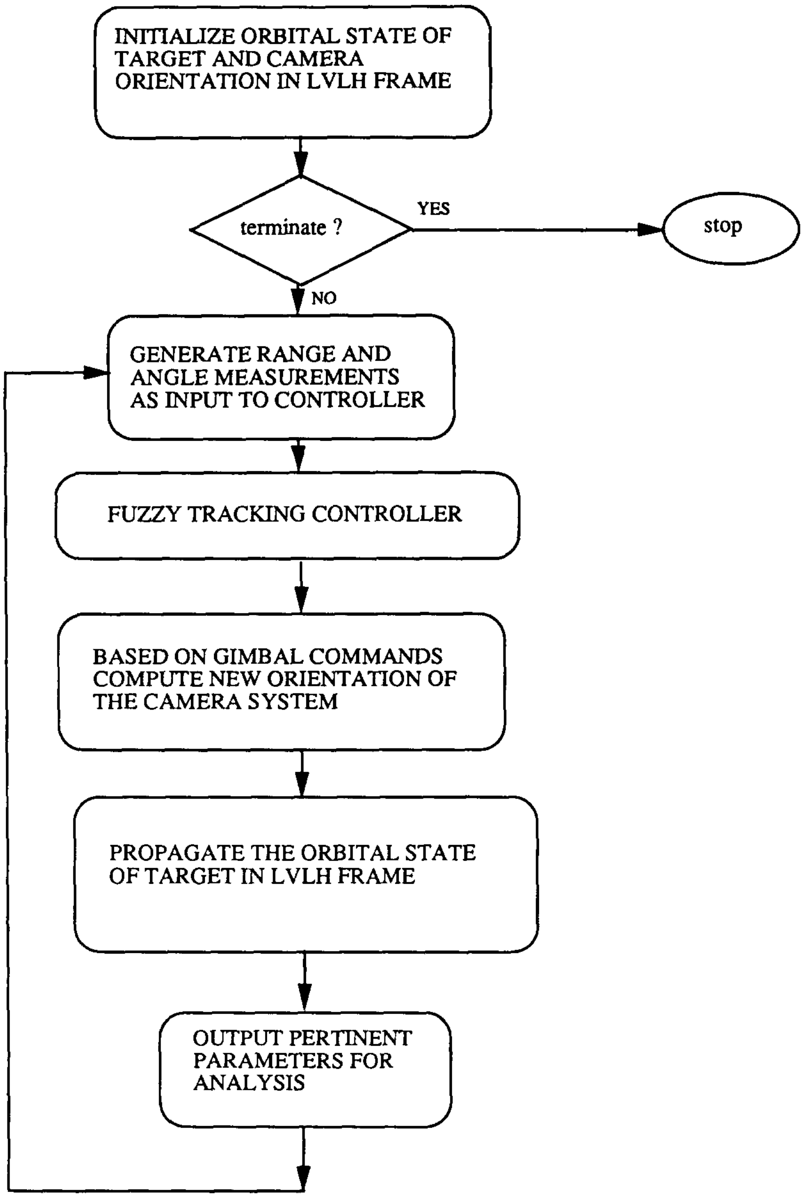


Figure 11. Testing of camera tracking fuzzy controller in simulation software.

A fuzzy logic approach to trajectory control has been developed (Lea et al. [36]) that allows the rover to avoid these hazards during the sample collection process. The fuzzy trajectory controller receives the goal or target point from the planner and uses x and y position errors as well as orientation (yaw) error in the control system frame and commands the rover in terms of steering angle and velocity. The fuzzy rule base containing 112 rules for the controller has been designed to drive the rover toward the x axis of the control error frame. As the rover approaches this axis, the rover is commanded to the correct orientation and then slowly drives toward the target point.

The x and y position error variables were modeled as a shouldered membership set of five piecewise linear functions (Hill et al. [19]) with a universe of discourse ranging from -100 to 100 m. The orientation or yaw error variable was modeled as an unshouldered membership set of seven functions with a universe of discourse ranging from $-180°$ to $180°$. The steering variable was modeled as an unshouldered membership set of five functions with a universe of discourse ranging from $-30°$ to $30°$. Finally, the velocity variable was modeled as an unshouldered membership set of seven functions with a universe of discourse ranging from -5 to 5 m/s.

A fuzzy trajectory controller for a Mars rover has been tested on several cases. Preliminary results have shown that the trajectory controller can reach the target position and attitude within 0.0005 on the x error axis, 0.25 m on the y error axis, with 0.45° yaw error. It is believed that these accuracies can be reduced by altering the membership function sets for the inputs and outputs. Further testing will facilitate the tailoring of the membership functions to the fuzzy rule set. Our activities in this project have shown that the fuzzy approach provides a control system that can be easily modified and tested.

6. FUTURE PROJECTS

In this section, the future activities that are planned for fiscal year 1991 and beyond are described with an expectation that these activities will be fully funded for new technology development. Activities in the area of traffic management around the space station *Freedom* (SSF) utilizing the camera tracking system are described first. Then the development of reinforcement learning during docking and repair operations is described. Development of a concept for a health monitoring system is described last.

6.1. Application of Camera Tracking System for Traffic Management

Future operations around SSF will include many vehicles approaching and departing the facility simultaneously. The crew on board SSF will have to perform traffic management functions very actively for safety reasons. The camera tracking system can be used effectively during these operations and can help the crew to efficiently manage traffic around SSF. During assembly and other extra vehicular operations, tracking and monitoring of other objects around SSF is required for mission success. As part of our future activities we will investigate the applicability of the camera tracking system to the problem of traffic management around SSF.

As part of our current activities we are planning to implement the fuzzy tracking controller in the hardware laboratory in the Engineering Directorate. The tracking controller will be interfaced with the gimbal drives and a pixel map from a camera. It is also possible to interface the output of the camera to a fuzzy hardware processor that can run the fuzzy controller and command the gimbal drives. It is planned to purchase suitable fuzzy hardware and perform the necessary testing to prove the concept at the hardware level. We will investigate the performance of fuzzy chips for accuracy, timing, and interfaces with a main computer. The use of the concept for several space station applications will be relatively easy and realizable.

The capabilities of the tracking controller can be expanded to perform other functions such as approach to the object, grapple, object identification, traffic management, and caution and warning to crew. Fast-moving objects can be identified easily via prediction of position, and thus collision avoidance can also be achieved. Since the system can work as a stand-alone system at the command level and will interrupt the operations flow only if necessary, it can become a node in a distributed sensor system.

6.2. Reinforcement Learning for External Environment During Docking and Repair Operations

A space shuttle crew initiates proximity operations procedures and docking maneuvers when the orbiter is within 1000 ft of the payload. It is expected that the payload will remain in a stable attitude and in nearly the same orbit during this entire time. Typically, the crew performs an approach known as the v-bar approach, keeping manual control of the orbiter. Docking maneuvers with the payload are also performed manually. The manual procedures and algorithms used during these tasks by the crew are developed using the real-time Shuttle Mission Simulator Facility on the ground.

During proximity operations, if the procedures require some adjustment, the so-called fine tuning, it is performed in real time even if it was not learned in the real-time simulation. Real-time adjustments are achieved based upon the current situation (e.g., satellite is not in a stable attitude or its orbit is constantly changing) and goal achievements. Thus the crew constantly learns and updates these procedures and algorithms as their experience base builds up.

It has been shown that a fuzzy logic controller can perform the same activities autonomously using sensor measurements as inputs. Fuzzy membership functions and the associated rule base (Lea [27, 28]) have been developed and integrated with the learning methods (Lee and Berenji [37]) developed at Ames Research Center (ARC) for controlling the inverted pendulum. The fuzzy controller can be combined with the reinforcement learning technique to give it a capability to learn in real time and improve its performance. With this capability, the fuzzy controller can adapt to a new environment and adjust its membership functions and/or rules to appropriately perform the tasks, given enough training instances.

The objectives of this project are to (1) combine the fuzzy controller developed for the translational motion with the reinforcement learning technique and (2) demonstrate its performance for the translational control of a spacecraft during proximity and docking operations. This project will be jointly undertaken by two NASA centers: JSC will provide a high-fidelity spacecraft simulation, test cases with input and output definitions, and preliminary rules and membership functions for the fuzzy translational controller, while ARC will provide the learning elements with appropriate interfaces to the simulation and updated rule base.

An approach has been developed (as shown in Fig. 12) to combine the fuzzy logic controller with reinforcement learning so that a higher level of autonomy for spacecraft operations can be achieved. Such an intelligent controller for a spacecraft is expected to adapt to the surrounding orbital environment and adjust its control strategy. Initial work on this project has been started, and a project plan has been put together (Lea et al. [38]). Details of the rule base, membership functions, input parameters, output commands, and other simulation interfaces are being worked out.

6.3. Concept Development for Health Monitoring System for Environment and Life Support System for Large-Volume Crew Quarters

Continuous monitoring and control of the Environment and Life Support System (ELSS) on board the space station *Freedom* (SSF) is required for the safety of the crew. The preliminary design of the ELSS control system (also known as atmospheric control system) consists of temperature, pressure, and composition controls, which are highly interrelated. The composition control includes control of major cabin atmosphere constituents, oxygen and nitrogen, and the control of humidity and trace contaminants. This preliminary design is based on the following requirements (Mankamyer [39]).

Relative humidity must be maintained between 25 and 70% with the constraint that the dew-point temperature is always maintained above 59°F.

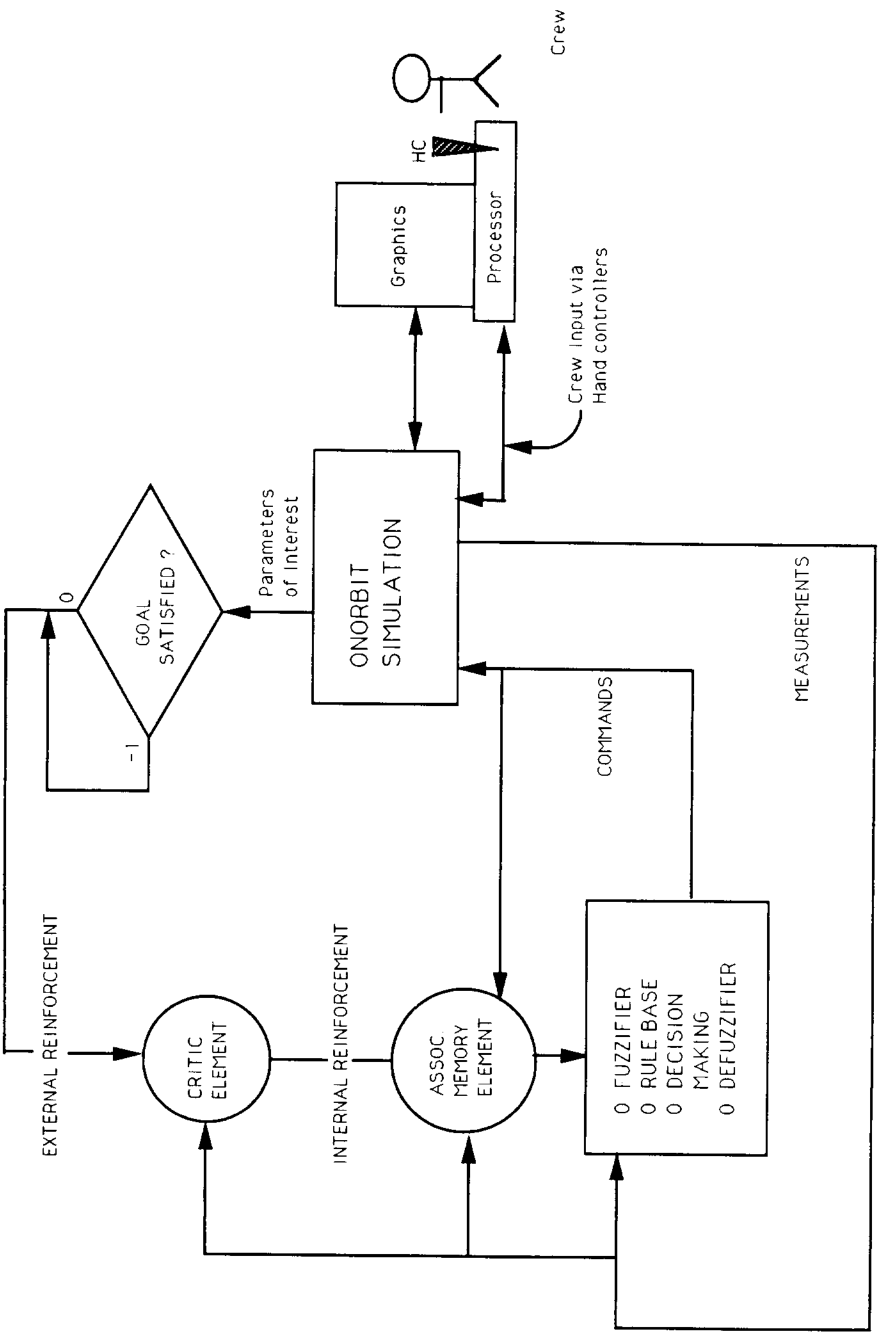


Figure 12. Fuzzy learning system for docking operations.

The cabin temperature must be selectable between 64 and 81°F and must be controlled within 1° accuracy. The cabin atmospheric pressure must be maintained at 14.7 psia within 0.2 psia accuracy. The oxygen partial pressure must be maintained at 2 psia.

The system dynamics model or the *plant* that represents the behavior of the system is nonlinear, and parameters are highly interrelated. The system equations can be linearized when the volume of the cabin is small, and several simplifying assumptions are made. However, the dynamics becomes increasingly complex and nonlinear as the volume of the crew quarters increases significantly. In such cases, applying conventional control theory will be very difficult, if not impossible.

In order to properly control the system state, highly accurate information regarding the current state of the system is required. Multiple sensor measurements are required to derive this accurate state information. It should be noted that the accuracy of state information is dependent on sensor accuracy. The sensors will possibly be distributed over the entire volume of the cabin. Thus, the problem can be thought of in two steps: deriving state information based on sensor measurements and controlling the deviations from the desired state. The first step relates to the interpretation of measurements, particularly their accuracy. The second step relates to the control of the state.

A concept of fuzzy logic based monitoring and diagnosis has been developed to combine several sensor measurements and derive the state information of a nonlinear system. The concept can be expanded to maintain a desired state, detect potential component failures, and generate immediate advisory messages for corrective action. As part of our activities in fiscal year 1991, we will apply this concept to the ELSS of SSF and implement a fuzzy rule base and membership functions. We will further generate a software demonstration as a proof of the concept and evaluate the suitability of the fuzzy logic based monitoring technique.

7. SUMMARY

Applications of fuzzy logic in autonomous orbital operations are described in this paper along with past accomplishments at JSC. Current ongoing as well as planned future activities are also described. The main objective of all these activities is to increase autonomy in orbital operation and thus achieve the higher level of operational efficiency desired for future space operations. The approach is to develop modular control that can be upscaled for greater autonomy in an integrated environment. The initial step is to develop a software controller and then to integrate it with hardware at the appropriate level. As the activities progress, detailed testing will be performed to check the implementation and integration of components. Our preliminary results promise a very successful utilization of fuzzy logic in autonomous orbital operations.

References

1. Zadeh, L., Fuzzy sets, *Inf. Control* **8**, 338–353, 1965.

2. Klir, G. J., and Folger, T. A., *Fuzzy Sets, Uncertainty, and Information*, Prentice-Hall, Englewood Cliffs, N.J., 1988.

3. Antsaklis, P. J., Passino, K. M., and Wang, S. J., An introduction to autonomous control systems, *Proc. 5th IEEE International Symposium on Intelligent Control*, **1**, 21–26, 1990.

4. Rogers, M., and Hoshiai, Y., The future looks "fuzzy," *Newsweek*, May 28, 1990.

5. Johnson, R. C., Clear leader emerges: Japan at fuzzy fore, *EETimes*, Sept. 11, 1989.

6. Armstrong, L., and Gross, N., Why "fuzzy logic" beats black-or-white thinking, Science & Technology section, *Business Week*, May 21, 1990.

7. Yasunobu, S., and Miyamoto, S., Automatic train operation system by predictive control, in *Industrial Applications of Fuzzy Control*, M. Sugeno, Ed., 1–18, North-Holland, Amsterdam, 1985.

8. Shingu, T., and Nishimori, E., Fuzzy based automatic focusing system for compact camera, *Proc. IFSA89*, 436–439, 1989.

9. Tobi, T., Hanafusa, T., Itoh, S., and Kashiwagi, N., Application of fuzzy control system to coke oven gas cooling plant, *Proc. IFSA89*, 16–22, 1989.

10. Yamakawa, T., How fuzzy microprocessors work, Tutorials at the International Workshop on Fuzzy System Applications, 79–87, 1988.

11. Togai, M., and Corder, R. J., A high speed fuzzy processor for embedded real-time applications, *Proc. SICE*, 1989.

12. Corder, R. J., A high speed fuzzy processor, *Proc. IFSA89*, 379–381, 1989.

13. Watanabe, H., VLSI chip for fuzzy logic inference, *Proc. 3rd International Fuzzy Systems Applications Congress*, 292–295, 1989.

14. Tasaka, Y., Hybrid bus type fuzzy controller with analog fuzzy chips, *Proc. IFSA89*, 280–283, 1989.

15. Johnson, R. C., Fuzzy faithful to get first glimpse of Expert, Technology section of *EETimes*, p. 25, June 5, 1989.

16. Lea, R. N., and Jani, Y. K., Automation issues for rendezvous and proximity operations, *Proc. Autonomous Rendezvous and Docking Conference*, NASA/Johnson Space Center, Houston, Texas, August 1990.

17. Togai, M., Commercial applications of fuzzy logic in Japan, Presentation at the Workshop on Fuzzy Control Systems and Space Station Applications (sponsored by

NASA/Ames Research Center and McDonnell Douglas Space Systems Company Space Station Division), Huntington Beach, Calif., November 1990.

18. Perkins, C., Teichrow, J., and Horstkotte, E., Fuzzy-C development system: a complete overview, Togai InfraLogic Inc., SOAR-89 conference held at Johnson Space Center, Houston, July 25-27, 1989.

19. Hill, G., Horstkotte, E., and Teichrow, J., *TIL Shell User's Manual*, release 1.1, Togai InfraLogic Inc., Irvine, Calif., January 1990.

20. Teichrow, J., and Horstkotte, E., *Fuzzy-C Compiler User's Manual*, version 2.0b, Togai InfraLogic Inc., Irvine, Calif., April 1989.

21. Lea, R. N., Giarratano, J., Fritz, R. H., and Jani, Y. K., Fuzzy logic control for camera tracking system, *Proc. 8th International Congress of Cybernetics and Systems*, New York, June 1990.

22. Lea, R. N., and Goodwin, M., Use of fuzzy logic to determine navigation filter convergence during space shuttle rendezvous, *ISA Proc. of Conf. on Robotics and Expert Systems*, NASA/Johnson Space Center, Houston, Tex., June 1986.

23. Lea, R. N., A fuzzy set approach to a navigation decision making problem, *Proc. 1985 Conference on Applied Analysis*, Univ. Houston at Clear Lake, Houston, Tex., 1985.

24. Lea, R. N., and Giarratano, J., An expert system program using fuzzy logic for shuttle rendezvous sensor control, *Proc. ROBEXS'86*, 327-329, 1986.

25. Lea, R. N., Applications of fuzzy sets to data monitoring and process control, *Proc. ISA88 Conf.*, **43**(4), 1495-1502, 1988.

26. Lea, R. N., Applications of fuzzy sets in redundancy management and filtering of sensor data, *Proc. Conference on Modeling and Simulation*, Univ. of Pittsburgh, **18**(3), 971-974, 1987.

27. Lea, R. N., Automated space vehicle control for rendezvous proximity operations, *Telematics and Informatics* **5**(3), 179-185, 1988.

28. Lea, R. N., Applications of fuzzy sets to rule-based expert system development, *Telematics and Informatics* **6**(3/4), 403-406, 1989.

29. Lea, R. N., Fuzzy logic space vehicle controller for rendezvous proximity operations, video conference demonstration from Johnson Space Center, International Workshop on Fuzzy Systems Applications (IFSA-88), Iizuka, Fukuoka, Japan, Aug. 20-24, 1988.

30. Lea, R. N., and Jani, Y., Spacecraft attitude control system based on fuzzy logic principles, *Proc. ROBEXS'89*, 1989.

31. Lea, R. N., Togai, M., Teichrow, J., and Jani, Y., Fuzzy logic approach to combined translational and rotational control of a spacecraft in proximity of the space station, *Proc. IFSA'89*, 23-29, 1989.

32. Lineberry, E. C., Lea, R. N., and Mitchell, E. V., Targeting equations and

control strategy for proximity operations, Internal Note, Mission Planning and Analysis Division, Johnson Space Center, March 1988.

33. Edwards, H. C., and Bailey, R., *The Orbital Operations Simulator User's Guide*, LinCom Corporation, ref. LM85-1001-01, June 1987.

34. Rockwell International, Clohessy-Wiltshire equations of motion, Shuttle targeting section, Shuttle onboard flight software subsystem requirements, *Guidance and Targeting*, **STS-83-0003b**, 1983.

35. Pal, S. K., Fuzziness, image information and scene analysis, in *An Introduction to Fuzzy Logic Applications in Intelligent Systems* (R. R. Yager and L. A. Zadeh, Eds.), Kluwer, Boston, 1992, (to appear).

36. Lea, R. N., Walters, L., and Jani, Y. K., A fuzzy logic approach to Mars rover trajectory planning and control, *Proc. 1st International Symposium on Measurement and Control in Robotics*, Session D.3, p. D.3.1.1, June 1990.

37. Lee, C. C., and Berenji, H. R., An intelligent controller based on approximate reasoning and reinforcement learning, *Proc. IEEE International Symposium on Intelligent Control*, Albany, N.Y., 1989.

38. Lea, R. N., Jani, Y. K., and Berenji, H., Fuzzy logic controller with reinforcement learning for proximity operations and docking, *5th IEEE International Symposium on Intelligent Control* **2**, 903, 1990.

39. Mankamayer, M., Atmospheric control systems, Presentation at the Workshop on Fuzzy Control Systems and Space Station Applications (sponsored by NASA/Ames Research Center and McDonnell Douglas Space Systems Company Space Station Division), Huntington Beach, Calif., November 1990.